



Haga lo que hicimos: autenticación API

¡Ahora está contigo! Realice el mismo procedimiento que hice en clase, implementando el proceso de autenticación en la API.

Opinión del instructor

Primero, deberá agregar Spring Security al proyecto, incluidas estas dependencias en el pom.xml:

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-security</artifactId>
</dependency>

<dependency>
  <groupId>org.springframework.security</groupId>
  <artifactId>spring-security-test</artifactId>
  <scope>test</scope>
</dependency>
```

[COPIA EL CÓDIGO](#)

Después de eso, deberá crear las clases *Usuario*, *UsuarioRepository* y *AutenticacionService* en el proyecto, como se muestra a continuación:

```
@Table(name = "usuarios")
@Entity(name = "Usuario")
```

@Getter

@NoArgsConstructor

@AllArgsConstructor

@EqualsAndHashCode(of = "id")

```
public class Usuario implements UserDetails {
```

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

```
private Long id;
```

```
private String login;
```

```
private String contrasena;
```

@Override

```
public Collection<? extends GrantedAuthority> getAuthorities()
```

```
    return List.of(new SimpleGrantedAuthority("ROLE_USER"));
```

```
}
```

@Override

```
public String getPassword() {
```

```
    return contrasena;
```

```
}
```

@Override

```
public String getUsername() {
```

```
    return login;
```

```
}
```

@Override

```
public boolean isAccountNonExpired() {
```

```
    return true;
```

```
}
```

@Override

```
public boolean isAccountNonLocked() {  
    return true;  
}  
  
@Override  
public boolean isCredentialsNonExpired() {  
    return true;  
}  
  
@Override  
public boolean isEnabled() {  
    return true;  
}  
}
```

[COPIA EL CÓDIGO](#)

```
public interface UsuarioRepository extends JpaRepository<Usuario,  
    UserDetails findByLogin(String login);  
}
```

[COPIA EL CÓDIGO](#)

```
@Service  
public class AutenticacionService implements UserDetailsService {  
  
    @Autowired  
    private UsuarioRepository repository;  
  
    @Override  
    public UserDetails loadUserByUsername(String username) throws UsernameNotFoundException {  
        return repository.findByLogin(username);  
    }  
}
```

```
}  
}
```

[COPIA EL CÓDIGO](#)

También debe crear una nueva migración en el proyecto para crear la tabla de usuario (IMPORTANTE: ¡recuerde detener el proyecto antes de crear la nueva migración!):

```
create table usuarios(  
    id bigint not null auto_increment,  
    login varchar(100) not null,  
    contrasena varchar(255) not null,  
  
    primary key(id)  
);
```

[COPIA EL CÓDIGO](#)

Además, también deberá crear la clase con la configuración de seguridad de la API:

```
@Configuration  
@EnableWebSecurity  
public class SecurityConfigurations {  
  
    @Bean  
    public SecurityFilterChain securityFilterChain(HttpSecurity http) {  
        return http.csrf().disable()  
            .sessionManagement().sessionCreationPolicy(SessionCreationPolicy.STATELESS)  
            .and().build();  
    }  
}
```

@Bean

```
public AuthenticationManager authenticationManager(Authent:
    return configuration.getAuthenticationManager();
}
```

@Bean

```
public PasswordEncoder passwordEncoder() {
    return new BCryptPasswordEncoder();
}
```

}

COPIA EL CÓDIGO

Finalmente, deberá crear una clase Controller y un DTO para manejar las solicitudes de autenticación en la API:

@RestController

@RequestMapping("/login")

```
public class AutenticacionController {
```

@Autowired

```
private AuthenticationManager manager;
```

@PostMapping

```
public ResponseEntity realizarLogin(@RequestBody @Valid Da
    var token = new UsernamePasswordAuthenticationToken(da
    var authenticaon = manager.authenticate(token);
```

```
    return ResponseEntity.ok().build();
```

```
}
```

```
}
```

COPIA EL CÓDIGO

```
public record DatosAutenticacion(String login, String contrasei  
}
```

[COPIA EL CÓDIGO](#)

Para probar la autenticación, deberá insertar un registro de usuario en su base de datos, en la tabla de usuarios:

```
insert into usuarios values(1, 'ana.souza@voll.med', '$2a$10$Y!
```

[COPIA EL CÓDIGO](#)